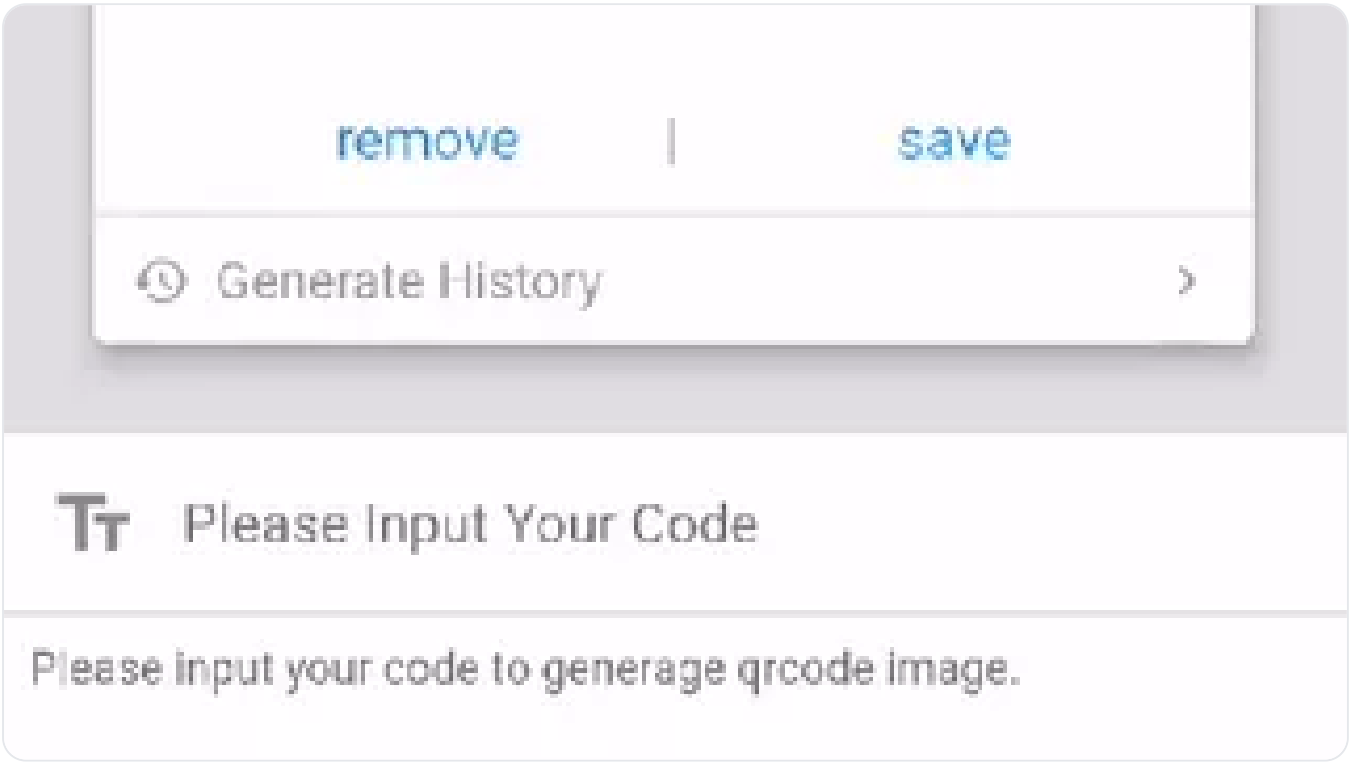
mobile_scanner

A Flutter plugin for scanning barcodes and QR codes across Android, iOS, macOS, and web.



About

mobile_scanner is a Flutter barcode and QR code scanner for apps that need camera-based code detection across multiple platforms. It uses CameraX and ML Kit on Android, AVFoundation and Apple Vision on iOS and macOS, and ZXing on the web. The package is built for fast, lightweight scanning with support for several barcode formats, real-time detection, and camera behavior controls such as scan windows and lens selection.

What is mobile_scanner

mobile_scanner is a Flutter barcode and QR code scanner plugin that lets developers

from scratch. It is designed for Android, iOS, macOS, and web, and the package notes explicit support for Flutter on those platforms while Linux and Windows are not supported.

The package combines native scanning stacks by platform: CameraX and ML Kit on Android, AVFoundation with Apple Vision on iOS and macOS, and ZXing on web. That makes it a cross-platform scanning layer rather than a single algorithm wrapped in a generic interface. The package description also emphasizes that it is fast and lightweight, which is useful if you need a practical scanner instead of a large computer-vision framework.

Key features

mobile_scanner focuses on the core tasks needed for barcode capture in production apps:

- Supports barcode and QR code scanning from the device camera.
- Detects multiple barcode formats rather than limiting you to one code type.
- Offers real-time detection for live camera scanning flows.
- Includes customization options for scanner and camera behavior.
- Supports image analysis on Android, iOS, and macOS.
- Can return images from detected barcodes on Android, iOS, and macOS.
- Supports a scan window on Android, iOS, and macOS for narrowing the active detection area.
- Supports auto-zoom on Android.
- Lets you query supported lens types on Android and iOS.
- Includes a helper for choosing the best close-range lens when the device supports it.

Feature support is not uniform across platforms. For example, scanWindow, analyzeImage, and returnImage are available on Android, iOS, and macOS but not on web. Auto-zoom is Android-only. Lens-type queries are available on Android and iOS, while macOS and web ignore the facing filter and return all lenses.

How to use it

Add the plugin to your Flutter project using the pub.dev package entry for mobile_scanner.

Check the platform you are targeting before relying on optional features like auto-zoom or image return.

Use the controller methods shown in the package examples to start scanning from the device camera.

If you want a tighter detection area, configure a scan window on supported platforms.

If your app needs better close-up scanning, call getSupportedLenses() and, when available, getBestCloseRangeScanningLens() before switching cameras.

A typical setup is to embed the scanner in a screen, then react to barcode results in real time. The docs point to runnable examples for basic usage, scan-window setup, and retrieving barcode images, so the example app is the best reference when you want to copy a supported pattern rather than guessing the API.

For lens handling, the package suggests using a facing filter when querying lenses on Android and iOS. That matters because without a facing filter, results may include both front and back cameras, which can lead to incorrect lens detection when users switch cameras.

Who it is for

This Flutter barcode scanner is a good fit for teams that need code reading in mobile or web apps and want one plugin across multiple targets. It is especially relevant for:

retail and checkout flows that need QR or barcode capture

ticketing and access apps that scan codes with the camera

inventory, logistics, or asset-tracking tools that read printed codes

Flutter developers who want a native-backed scanner instead of a fully custom camera stack

apps that need platform-aware behavior, such as lens selection or scan windows

It is also a strong match when you need a package with documented platform support instead of a generic camera view. Because the plugin already lists which features work on each platform, it is easier to plan around known limitations before shipping.

What to know before you use it

mobile_scanner is useful, but it is not a universal feature set on every platform. Web support exists, but some camera controls and image-analysis features are unavailable there. Linux and Windows are listed as unsupported, so this is not the right choice if desktop coverage on those operating systems is a requirement.

The package also does not promise that every camera-lens query can be selected directly on every platform. On Android, the docs note that physical sub-cameras are not independently selectable through CameraX in the same way you might expect from a full hardware picker, so some results are best treated as guidance rather than direct guarantees. On iOS 15+, the close-range lens helper can choose the camera with the shortest minimum focus distance, but on older iOS versions and on macOS it falls back to a normal lens.

For code that depends on detection accuracy, always test with your own barcode formats, lighting conditions, and devices. If you need authoritative device camera behavior, hardware capabilities, or platform-native edge cases, use the official platform documentation alongside this plugin. The package is a scanning layer, not a substitute for native camera APIs or device-specific validation. For production use, confirm behavior against the app's supported platforms and the package examples before you rely on it in a release build.

In short, mobile_scanner is a cross-platform barcode scanning plugin for Flutter that saves setup time, but you should still verify platform-specific feature gaps before depending on it in critical workflows.

FAQ

Is mobile_scanner free to use?

Which platforms does mobile_scanner support?

Does mobile_scanner work with QR codes only?

Can I use mobile_scanner for both QR codes and barcodes?

Can I use mobile_scanner for QR codes and barcodes?

Can I control the scanner window or camera behavior?

Does mobile_scanner work on the web?

How do I choose the best camera lens for close-up scanning?

Does mobile_scanner work with Flutter on the web?

Who should use mobile_scanner?

When should I use getBestCloseRangeScanningLens()?

How do I choose the best camera lens for close-up scans?

Sponsored

Belay – Awa... Featured

Developer Tools

Stays awake while your agents work.

Free # Developer Tools

View ↗

Partly Featured

Finance

Partly is a mobile app that lets friends quickly scan receipts, split any bill fairl...

Free # Finance

View ↗

mobile-scanner

Pricing	Free
Category	Developer Tools
Platform	Android, iOS, macOS, Web
Launched	Aug 2026
Updated	today
Verified	4 days ago

TAGS

flutter # barcode-scanner # qr-code # camera # mobile-development # cross-platform # web-scan # pub-dev

© 2026 Dir Hub